DEPARTMENT OF DEFENSE

HIGH PERFORMANCE COMPUTING MODERNIZATION PROGRAM

FY 2002

HPC SYSTEM PERFORMANCE METRICS

NOVEMBER 2001

TABLE OF CONTENTS

DoD High Performance Computing Modernization Program HPC System Performance Metrics

1. INTRODUCTION AND GENERAL POLICY

The HPC system performance metrics detailed in this document are an important subset of the overall program performance metrics collected, aggregated, and analyzed by the High Performance Computing Modernization Office (HPCMO). This document does not address cost and schedule metrics, nor does it address other performance metrics, such as user help desk statistics and user satisfaction metrics. It also does not address detailed utilization reports produced by the shared resource centers for their user organizations.

HPC system usage and job turnaround times at the centers are key indicators of the usefulness of High Performance Computing Modernization Program's (HPCMP) resources to DoD's science and technology (S&T) and test and evaluation (T&E) communities. We use system performance metrics to make resource allocation decisions, plan future resource acquisitions, and validate user requirements. Although maximum throughput is not always the major goal for HPC resources, accurate system performance data is essential for HPCMP strategic planning.

One principal concern of most users is how quickly a computation can be run in a multi-user environment. For this reason, the HPCMO has been collecting usage and turnaround time data on its HPC systems for the past several years at the user organization and project level. By evaluating turnaround time metrics defined in this document, we are able to measure how effectively we are supporting our customer base.

The HPCMP began to collect usage data for all HPC systems at each of its shared resource centers by computational project in FY98. The purpose of this paper is to define the set of system performance metrics that will be collected and how they will be reported in the future. A common set of system performance metrics in a uniform format will be obtained from each MSRC and all distributed center systems that address non-real-time needs. For HPC systems that address both real-time (RT) and non-real-time (NRT) needs, a separate set of metrics will be kept for each of these two activities. Metrics for the NRT workload on such a system will be identical to those for the NRT systems at the MSRCs and distributed centers. For real-time activities, in addition to a subset of the aforementioned metrics, test or simulation activities performed will be reported as discrete events within a computational project.

2. SPECIFIC SYSTEM PERFORMANCE METRICS

2.1 CPU USAGE

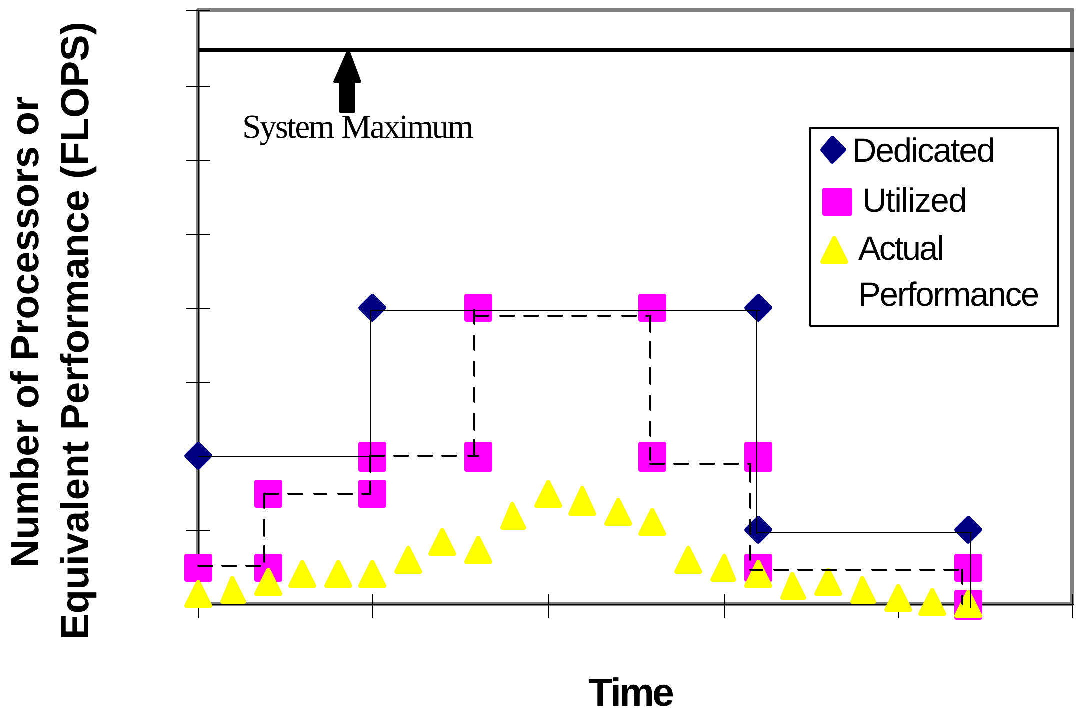
The primary goal of collecting system performance metrics is to track CPU usage for each computational project. CPU usage for each job (or job segment) will be determined by multiplying the number of dedicated processors for a job by the number of hours the job occupies those processors. For this document, job segment is defined as any part of a job that utilizes a constant number of processors. Thus, usage will be computed as dedicated wall clock hours dedicated to a particular job (or job segment) while that job is in execution. Total usage for a computational project will be obtained by summing the usage charged to each job (or job segment) for that project. This total dedicated wall clock usage will be reported for each computational project for each reporting period. Total dedicated wall clock usage for each project will be compared with that project's allocation, since this quantity represents CPU resources unavailable to all other projects using that HPC system. Usage to be charged to a project's allocation and usage resulting from use of the background queues will be reported separately for each project.

For NRT systems that allow non-exclusive use of processors in a time-sharing mode, actual CPU usage time, rather than dedicated wall-clock time, will be compared with a project's allocation. Figure 1 plots CPU usage for an NRT system vs. time for a typical job; the area under the solid line represents wall clock (dedicated) usage for that job. The area enclosed by the solid line and the line representing the maximum number of processors available on that system represents CPU resources available to all other projects using that HPC system. Note that some systems are not capable of changing the number of dedicated nodes during a job, in which case the dedicated number of processors remains constant. Usage on RT systems will be reported as dedicated wall clock usage when these systems are employed in RT mode.

CPU usage as defined above will be reported for each user computational project as well as for any system or special accounts, such as 99xx accounts. Note that no usage should accrue against reserve (9000-9199) accounts since these accounts are provided to each organization as a mechanism to reallocate HPC resources and not for actual usage. In addition, each center will report down time in CPU-hours in the following categories: scheduled maintenance, unscheduled down time, and time unavailable for other reasons. The list of test and special accounts to be used to report special types of usage and down time is given in Attachment 1.

Some systems have software capable of reporting actual CPU usage (in terms of total number of hours a CPU is actually engaged during a job or job segment) as well as dedicated wall clock usage. For those systems, actual CPU usage and dedicated wall clock usage for the representative subsets of jobs run on that system may also be determined and reported. Comparison of actual CPU usage to dedicated CPU usage as measured by wall clock times will allow assessments of the effectiveness of system software in scheduling blocks of processors to maximize overall throughput in combination with the effectiveness of applications software in terms of load balancing. Referring again to Figure 1, the area under the dashed line represents actual CPU usage for that job. As hardware performance monitors become available on various systems at each shared resource center, that shared resource center may work with the HPCMO to define sets of user jobs for which hardware performance statistics, such as actual numbers of FLOPS performed, are gathered and reported. Comparison of actual system performance for a job to peak performance of the block of processors being used by that job will indicate the effectiveness of application software in its usage of that HPC system. The triangle symbols in Figure 1 represent actual performance of a given job in terms of FLOPS performed.

Figure 1. Individual job usage



Each shared resource center is asked to assess progress towards multiprocessor usage by preparing histograms, on a quarterly basis, profiling CPU assignments for all jobs on each of its HPC systems capable of producing the necessary data. The histograms will plot total numbers of jobs and total numbers of CPU hours for each number of CPUs assigned over all jobs run on that system for each quarter. Over time, the resultant profile will document progress towards multiprocessor usage and capability utilized on each system, as opposed to simply measuring total throughput. Each shared resource center manager may propose additional metrics for documenting multiprocessor usage and required capability.

2.2 Number of Users

For NRT activities, the number of HPC users on a particular HPC system will be provided in two ways:

(1) The number of active users will be reported as the number of active accounts that show at least one hour of CPU usage for the reported month.

(2) The cumulative number of users for a fiscal year will be reported as the number of active accounts that had at least one hour of CPU usage for any month within the current fiscal year.

The number of users (both active and cumulative) will be reported for each computational project for each reporting period. The number of users for a RT activity will include a count of all personnel directly involved in the operation of the HPC system during the test or simulation activity. These users will also be reported for each month and cumulatively for the fiscal year.

2.3 EXPANSION FACTORS

Job turnaround time for NRT systems may be captured in terms of the expansion factor, which is defined for an individual job (or job segment) as the total elapsed time from job submission to job completion (see Figure 2), divided by the total time that job (or job segment) is actually executing (Equation 1).

$$\mathbf{EF_i = (QWT_i + WCT_i) / ET_i} \quad \mathbf{(1)}$$

EF_i = expansion factor for job i
QWT_i = queue wait time for job i
WCT_i = wall clock time (time from beginning to end of execution for job i)
ET_i = system execution time (total time job i is actually being executed)

The execution time, ET_i, excludes any execution disruption times such as system down time or the time the job is check-pointed. In the case of jobs running with multiple processors, the total execution time for the relevant segments is obtained by multiplying execution time per processor by the number of processors associated with the segment. The wall clock time (previously referred to as execution wall clock time in our documents) includes the total length of time between the beginning of job execution and the completion of the job, and includes checkpoints, restarts, and system down times within the time period of the total job. Note that for interactive jobs on non-real-time systems, the queue wait time is zero so the expansion factor simplifies to wall clock time divided by system execution time. This expansion factor EF_i is also termed an un-normalized expansion factor (UEF_i) if the number of processors used is not taken into account as in Equation 1. Since CPU usage for the job (or job segment) is defined as the number of processors dedicated to that job multiplied by the time those processors are dedicated (system execution time), the denominator of the expression for the expansion factor can be replaced with CPU usage for that job, provided the numerator is also multiplied by the number of processors (Equation 2). The expansion factor in this case is named the normalized expansion factor (NEF_i).

$$\mathbf{NEF_i = n_i\,(QWT_i + WCT_i) / CPU_i} \quad \mathbf{(2)}$$

n_i = number of dedicated processors for job i

CPU_i = CPU usage for job i

Equation 2 will reduce to Equation 1 when the job uses only a single processor.

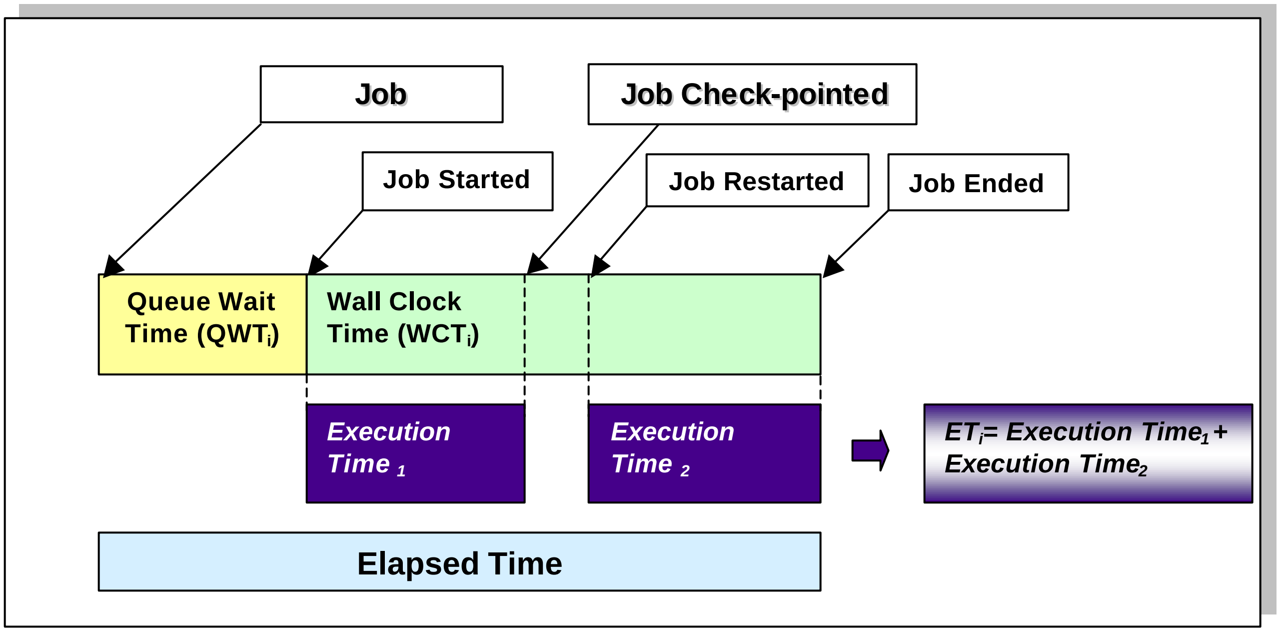


Figure 2. Computational Process for a Job

The expansion factor has the property that its minimum (optimum) value is always one, independent of the number of processors.

If the number of processors n_i is removed from Equation 2, the resultant expansion factor is unnormalized and loses the property that its minimum value is one. The minimum of this unnormalized expansion factor is the reciprocal of the number of processors dedicated to that job. The reciprocal of the unnormalized expansion factor, termed the effective number of processors, can be interpreted as the average number of processors utilized during the entire job time, from job submission to job completion.

Calculation of a normalized expansion factor (NEF) for a group of jobs must be done by first calculating NEFs for all individual job segments and then performing a weighted average over all job segments in the group, using the CPU usage for each individual job segment as its weighting factor. Use of the CPU usage as the weighting factor in calculating the average ensures that each job segment's contribution to the expansion factor will depend on its CPU time. Equation 3 illustrates the calculation of this weighted average of the NEF for a group of job segments.

$$\overline{\mathbf{NEF}}_{\mathbf{CPU}} = S_{\mathbf{i}}\ (\mathbf{CPU}_{\mathbf{i}} \cdot \mathbf{NEF}_{\mathbf{i}})\ /\ S_{\mathbf{i}}\ \mathbf{CPU}_{\mathbf{i}} \qquad \mathbf{(3)}$$

$\overline{NEF}_{CPU}$ = CPU usage-weighted average of the normalized expansion factor
S_i = summation over all job segments in the group to be averaged

In a similar way we can also calculate the CPU usage-weighted average of the un-normalized expansion factor ($\overline{UEF}_{CPU}$). Algebraically, it can be shown that the CPU usage-weighted average of the un-normalized expansion factor can be calculated from the averages of CPU usage, queue wait time, and execution wall clock time, as shown in Equation 4.

$$\overline{\mathbf{UEF}}_{\mathbf{CPU}} = (\langle QWT \rangle + \langle WCT \rangle) / \langle CPU \rangle \qquad \textbf{(4)}$$

$\overline{UEF}_{CPU}$ = CPU usage-weighted average of the unnormalized expansion factor
<QWT> = average queue wait time
<WCT> = average execution wall clock time
<CPU> = average CPU usage

2.4 EXPANSION FACTOR REPORTS

A summary expansion factor report for each class of jobs will be provided monthly. This summary will contain the class of jobs (see Figure 3), number of runs in the class, average queue wait time, average wall clock time, average number of CPUs for the jobs in the class, in addition to CPU-weighted averages of un-normalized expansion factors and normalized expansion factors for each class of jobs. Classes of jobs include background, Challenge, Service/Agency high priority, Service/Agency standard, and urgent. Explanations of the background, Challenge, high priority, standard, and urgent class of jobs is given in Figure 3. A sample summary expansion factor report is attached (Attachment 2). Reports will be submitted by the 15th of each month and along with the reports, explanations should be provided for any expansion factors that exceed the target for that classification. The HPCMO may request a job-by-job detailed expansion factor report if additional explanation is necessary. A sample job-by job detailed expansion factor report is presented in Attachment 3.

2.5 COMMERCIAL-OFF-THE-SHELF (COTS) SOFTWARE USAGE METRICS

Usage of COTS software packages is an important metric to monitor to ensure that COTS software being provided by a center meets users' needs. Each center will report usage on a targeted list of these COTS software packages by dedicated CPU-hours used on and the number of accesses of those packages when requested.

Computational Work at the Shared Resource

All computational work is to be authorized by the applicable S/AAA and administered at the centers.

Background Jobs:

1. This category is for computational work done through background-only accounts, Challenge accounts, or Service/Agency accounts
2. Runs at lowest priority
3. No appreciable impact on allocated hours
4. No target expansion factor

High-Priority Service/Agency Jobs:

1. Time critical Service/Agency jobs that occur on a regular or recurring basis. Short turnaround time is required for these jobs.
2. Allocated by Service/Agency. Allocation comes out of existing Service/Agency allocations.
3. Has to be approved by Service/Agency principal
4. HPCMO must be notified.
5. Runs at high priority
6. Target expansion factor is 1.7.

Challenge Jobs:

1. Challenge Projects selected by DUSD (S&T)
2. Allocated by HPCMO
3. Re-allocation done by HPCMO
4. Runs at high priority
5. Target expansion factor is 1.7.

Urgent Jobs:

1. Single time-sensitive event that results from unexpected need. This job needs faster-than-normal turnaround on computational resources.
2. Allocated by the Service/Agency. Allocation comes out of the Service/Agency allocations
3. Approved by HPCMP Director
4. Has to be approved by HPCAP if the Service/Agency cannot allocate the requested time
5. Runs at highest priority
6. Target expansion factor is 1.1.

Standard Service/Agency Jobs:

1. Computational work as allocated through Service/Agency allocation process
2. Allocated by the Service/Agency
3. Runs at standard priority
4. Target expansion factor is 2.2

Figure 3. Description of the Job Categories

2.6 OTHER FUTURE USAGE METRICS

The HPCMO realizes that CPU usage is not the only metric indicative of efficient and effective HPC system use. In particular, efficient memory usage is also critical to the overall effectiveness of an HPC system to address important applications. One such new metric, "memory-time integral", has been found to be useful and is included in the detailed job-by-job expansion factor report. Each shared resource center manager may propose potential memory usage metrics for HPC systems at that shared resource center and a suggested time frame for implementation. In addition, combined metrics (such as CPU usage coupled with memory usage) may also be proposed. Input/output operations are also critical to effective operation of an HPC system. Each shared resource center manager may propose potential measures of input/output operations for HPC systems at the shared resource center and a suggested time frame for implementation.

2.7 ADDITIONAL REAL-TIME METRICS

Each shared resource center with mixed real-time and non-real-time systems will compute all of the non-real-time metrics discussed above for the non-real-time portion of that system's operation. Of the previously discussed system performance metrics, only dedicated wall clock CPU time and numbers of users will be reported for the real-time portion of each system's operation, including those systems operated exclusively in real-time mode. In addition, a listing and the duration of test and simulation real-time activities performed will be reported for each computational project and test project supported by the real-time operations of each HPC system. These real-time activities include dedicated simulation events and demonstrations run on MSRC systems. For mixed systems, all applicable system performance metrics will be independently tracked for each activity (non-real time and real time) and reported separately.

3. TRACKING AND REPORTING OF SYSTEM PERFORMANCE METRICS

A separate system performance metrics reporting spreadsheet for CPU usage will be prepared (Attachment 4). All system performance metrics, unless otherwise specified, will be reported monthly by each shared resource center to the HPCMO as part of its monthly reporting requirements. The HPCMO will maintain usage databases and spreadsheets capable of rolling up total usage for each computational project, user organization, and Service/Agency by system type or total across-the-board usage in normalized usage units of gigaFLOPS-years. The HPCMO will issue periodic system performance reports to all of its shared resource centers and user organizations. Attachment 5 gives a summary list of system performance metrics discussed in this paper.

4. CONCLUSION

The HPCMO will work with its shared resource centers to continue development and implementation of system performance metrics that are vital to the effective monitoring of the success of the HPCMP. These metrics can provide effective measures of progress in hardware, system software, applications software, and the overall ability of DoD HPC users to take full advantage of the tremendous capabilities and capacities provided by the program.

ATTACHMENTS

Attachment 1

CHSSI Test Accounts

Project	Account	Status (Estimated Completion)
CCM - 05	9948	2003
CCM - 06	9949	2003
CEA - 08	9950	2003
CEA - 09	9951	2003
CEA - 10	9952	2003
CEN - 05	9953	2003
CSM - 05	9955	2003
CWO - 04	9956	2003
CWO - 05	9957	2003
CWO - 06	9958	2003
CWO - 07	9959	2003
EBE - 01	9964	2003
EBE - 02	9964	2003
EBE - 03	9964	2003
FMS - 05	9936	2001
FMS - 06	9960	2003
FMS - 07	9966	2003
HIE - 01 thru 06	9965	2003
IMT - 02	9945	2001
IMT - 04	9947	2001
IMT - 05	9961	2003
MBD - 01 thru 03	9969	2006
SIP - 06	9931	2003
SIP - 07	9962	2003
SIP - 08	9963	2003
SLE - 01A thru 04A	9967	2003
SOS - 01-thru 03	9970	2006
SPG - 01 thru 05	9971	2006
WTI - 01 thru 03	9968	2003

Special Accounts

Type	Account
Support	9999
S/AAA	9998
PET	9997
Training	9996
Outreach	9995
Meta-Center Projects	9994
Scheduled Maintenance	9993
Unscheduled Down Time	9992
Time Unavailable for Other Reasons	9991
Benchmark Activities	9990

Attachment 2

Month	July 01	Center	ERDC	System	Cray T3E	Maximum number of Processors	512
Class	Number of Runs	Average Queue Wait Time (Hrs)	Average Wall Clock Time (Hrs)	Average CPU Time (Hrs)	Average # CPUs	Un-Normalized Expansion Factor	Normalized Expansion Factor
Background	183	1.476	2.101	126.562	52.144	0.028	1.550
Challenge	944	1.432	3.272	241.183	58.705	0.019	1.473
High Priority	10	0.010	6.000	385.000	125.600	0.022	1.350
Standard	1127	1.626	3.149	317.137	67.480	0.015	2.055
Urgent	2	1.000	3.000	284.700	22.400	0.015	2.000
All Jobs	2,266	1.205	3.257	301.245	63.211	0.017	1.886

Overall expansion factor: Overall normalized expansion factor = sum of (normalized expansion factor * CPU time) for each job / sum of CPU times for all the jobs.

Class: jobs category belong to Background (B), Challenge (C), High-Priority (H), Standard (S), Urgent (U), or other (O) class.

Number of runs: Total number of runs for each category. In the example there were 183 background jobs

Average Queue Wait Time in Hours: For each job, queue wait time (QWT) is the difference in job starting time and the time of submission of the job. Compute the average of the QWT for the class of jobs under consideration.

Average Wall Clock Time in Hours: For each job, wall clock time (WCT) is the difference between job starting time and the job ending time. This will include system down times, check point time and restart time. Compute the average of the WCT for the class of jobs under consideration.

Average CPU time: Average job execution time for the class of jobs under consideration. Sum of all the CPU times/number of jobs

Average # CPUs: Average number of CPUs for the class of jobs under consideration.

Un-normalized Expansion Factor: Calculate the un-normalized expansion factor for each class of jobs. Un-normalized expansion factor for each job = (QWT+WCT)/CPU time. Un-normalized expansion factor for a class of jobs is the weight-averaged un-normalized expansion factor. Use CPU time as the weight. Un-normalized Expansion factor= Sum of (Un-normalized expansion factor* CPU time)/ sum of CPU times of the jobs in the class.

Normalized Expansion Factor: Calculate the normalized expansion factor for each job. Normalized expansion factor = number of processors* (QWT+WCT)/CPU time. Normalized expansion factor for a class of jobs is the weight-averaged normalized expansion factor: Use CPU time as the weight. Normalized expansion factor = sum of (normalized expansion factor* CPU times)/sum of CPU times of the jobs in the class.

Attachment 3

Note: This detailed usage report should be sent only if specifically requested by the HPCMO

Month	July 01		Center	ARL	System	SGI Origin 3000	Maximum number of Processors	256				
Request #	Project	Class	Service	Queue	Wall Clock Time (secs)	Queue Wait Time (Secs)	Elapsed Time (Secs)	CPU Time (Secs)	Number of Processors (NCPUs)	Unnormalized Expansion Factor (UEF)	Normalized Expansion Factor (NEF)	Memory time Integral (GB*Secs)
364022	NRLMR03794C57	C	Navy	zornig2_pe_12hr_gc.q	32844.000	7.000	32851.000	1313760.000	40	0.025	1.000	81772.900
364076	NRLMR03794C57	C	Navy	zornig2_pe_12hr_gc.q	18594.000	28.000	18622.000	743760.000	40	0.025	1.002	23033.800
349972	DSWAA02472C50	C	Navy	zornig2_pe_48hr_gc.q	3.000	7145.000	7148.000	120.000	40	59.567	2.000	0.017
349974	DSWAA02472C50	C	Navy	zornig2_pe_48hr_gc.q	2.000	7097.000	7099.000	80.000	40	88.737	3.000	0.011
349975	DSWAA02472C50	C	Navy	zornig2_pe_48hr_gc.q	3.000	7118.000	7121.000	120.000	40	59.342	2.000	0.017
349976	DSWAA02472C50	C	Navy	zornig2_pe_48hr_gc.q	3.000	7161.000	7164.000	120.000	40	59.700	2.000	0.017
350076	NRLDC04382255	S	Navy	zornig_pe_12hr.q	1470.000	6.000	1476.000	58800.000	40	0.025	1.004	823.398
351717	HPCMO9947AIM4	O	Other	zornig_pe_4hr.q	291.000	724.000	1015.000	11640.000	40	0.087	3.488	208.792
364902	ARLAP02642C47	C	Army	zornig_pe_24hr_gc.q	719.000	7.000	726.000	30198.000	42	0.024	1.010	0.000
364903	ARLAP02642C47	C	Army	zornig_pe_24hr_gc.q	59.000	6.000	65.000	2478.000	42	0.026	1.102	114.618
364904	ARLAP02642C47	C	Army	zornig_pe_24hr_gc.q	104.000	5.000	109.000	4368.000	42	0.025	1.048	0.000
364905	ARLAP02642C47	C	Army	zornig_pe_24hr_gc.q	2819.000	8.000	2827.000	118398.000	42	0.024	1.003	0.000
364333	ARLAP02642C47	C	Army	zornig2_pe_24hr_gc.q	6740.000	20.000	6760.000	283080.000	42	0.024	1.003	1390000.000
364425	ARLAP02642C47	B	Army	zornig2_pe_24hr_gc.q	5065.000	5733.000	10798.000	212730.000	42	0.051	2.132	48970.300
353253	HPCMO9947AIM4	O	Other	zornig2_pe_4hr.q	357.000	91.000	448.000	10710.000	30	0.042	1.255	360.677
353275	HPCMO9947AIM4	O	Other	zornig2_pe_4hr.q	516.000	2583.000	3099.000	15480.000	30	0.200	6.006	211.530
352001	HPCMO9947AIM4	O	Other	zornig2_pe_4hr.q	4049.000	118.000	4167.000	125519.000	31	0.033	1.029	1270.540
352006	HPCMO9947AIM4	O	Other	zornig2_pe_4hr.q	1947.000	1233.000	3180.000	60357.000	31	0.053	1.633	30.689
352701	ARLAP99990999	B	AF	zornig_bmark.q	18776.000	258.000	19034.000	600832.000	32	0.032	1.014	1290000.000
353332	ARLAP99990999	S	DOTE	zornig_bmark.q	21681.000	13720.000	35401.000	693792.000	32	0.051	1.633	1490000.000
361906	ARLAP99990999	S	DARPA	zornig_bmark.q	14225.000	14585.000	28810.000	455200.000	32	0.063	2.025	986384.000
350102	ARLAP99990999	H	DTRA	zornig_pe_12hr.q	27498.000	24.000	27522.000	879936.000	32	0.031	1.001	0.000
350881	ARLAP99990999	S	BMDO	zornig_pe_12hr.q	14642.000	19.000	14661.000	468544.000	32	0.031	1.001	1020000.000
350943	ARLAP99990999	S	Army	zornig_pe_12hr.q	16779.000	19.000	16798.000	536928.000	32	0.031	1.001	1160000.000
351305	ARLAP99990999	S	Army	zornig_pe_12hr.q	12405.000	19.000	12424.000	396960.000	32	0.031	1.002	855816.000
355015	ARLAP99990999	S	Army	zornig_pe_12hr.q	21647.000	20.000	21667.000	692704.000	32	0.031	1.001	1490000.000
357236	ARLAP14874100	U	Army	zornig_pe_12hr.q	258.000	6056.000	6314.000	8256.000	32	0.765	24.473	115.036

Attachment 4

Service	SRC ID	HPC System ID	User Organization	Computational Project Number	CTA Number	Allocation Request Number	Background	CPU Hours	Allocation	# of Active Users	# of Cumulative Users
	16-character identification code										
AF	AB	1	AFOSR	1234	3			222	10000	1	3
DTRA	AK	2	DTRAA	5678	3		B	125	15000	1	2
Navy	ST	4	ONRDC	4321	3	C88		16	500	1	1

Example:

SRC ID	First project ran at ARL
	Second project ran at ARSC
	Third project ran at NAVO

HPC System ID	First entry used the Cray SV-1 with 28 processors
	Second entry used the Cray T3E with 272 processors
	Third entry used the SGI Origin 2000 with 52 processors

User Organization and Compuational Project Number	First entry is associated with project number 1234 at AFOSR
	Second entry is associated with project 5678 at DTRA
	Third entry is associated with project 4321 at ONR

CTA Number	All three projects are doing Computational Chemistry and Materials Science

Allocation Request Number	Third entry is a Challenge Project

Background	The second entry ran its job in the background

CPU Hours	First entry is the number of hours run on the Cray SV1 at ARL for the "1234" project during the month reported
	Second entry is the number of hours run on the Cray T3E at ARSC for the project "5678" during the month reported
	Third entry is the number of hours run on the SGI Origin at NAVO for the project "4321"during the month reported

Allocation	First entry is the number of hours allocated for this project on the specific ARL Cray SV1 during FY 2002
	Second entry is the number of hours allocated for this project on the specific ARSC Cray T3E during FY 2002
	Third entry is the number of hours allocated for this project on the specific NAVO SGI Origin during FY 2002

# of Active Users	First entry shows that the project had one active user on the Cray SV1 during the month reported
	Second entry shows that the project had one active user on the Cray T3E during the month reported
	Third entry shows that the project had one active user on the SGI Origin during the month reported

# of Cumulative Users	To date, the first project has had three distinct users running jobs on this system - no person is counted twice!
	To date, the second project has had two distinct users running jobs on this system - no person is counted twice!
	To date, the third project has only had one user running jobs on this system

Attachment 5

Specific System Performance Metrics to Be Reported

Monthly Metrics

1. Project-level usage for each computational project, including DoD Challenge Projects[1]
2. Number of active and cumulative users for each computational project[1]
3. An expansion factor summary report giving details of class of jobs, number of runs in the class, average queue wait time, average wall clock time, average CPU time and average number of processors of the jobs in the class, and CPU-weighted averages of un-normalized and normalized expansion factors for each class of jobs
4. Expansion factor for each DoD Challenge Project on each HPC system
5. System time in CPU-hours for each system, categorized into scheduled maintenance, unscheduled down time, and unavailable for other reasons (see Attachment 4)
6. Separation of all system performance metrics between real-time and non-real-time activities for systems that have both kinds of workloads
7. A listing of each discrete test or simulation performed in a real-time mode

Quarterly Metrics

8. Histograms showing distribution of workload by number of processors, including number of jobs and total CPU time

Occasional Metrics

9. Actual CPU time, as compared with dedicated wall clock usage
10. Hardware performance monitoring
11. Job-by-job detailed expansion factor report

Additional Future Metrics to Be Defined

12. Additional metrics for documenting multiprocessor usage
13. Memory usage metrics
14. Input/output and/or storage metrics
15. Combined metrics

[1] Project-level data to be electronically reported separately by usage counted against allocations and background usage not counted against allocations (with a "B" in the background column) in an Excel spreadsheet (see Attachment 4).

Metrics for RT systems and RT activities on mixed systems will include all metrics except 3, 4, and 11 listed above. Metrics for NRT systems and NRT activities on mixed systems will include all metrics listed above.